REMARKS / ARGUMENTS

Claims 1-26 have again been rejected under 35 USC 102(e) as being anticipated by Habegger (US Pat. No. 6,643,642 B1) ("Habegger"). Habegger clearly states in the "FIELD OF THE INVENTION" that "the invention relates generally to the field of databases". By contrast, as stated in Applicant's "FIELD OF THE INVENTION", Applicant's invention relates to "graphical user interfaces". Although each employs a hierarchical data structure, each invention uses it in a different way, with different additional elements, and for a different purpose.

Habegger uses the hierarchical data structure to facilitate searching of the database (see Field of the Invention). By contrast, Applicant uses the hierarchical data structure to facilitate display of a plurality of machine vision elements (See Summary of the Invention). The hierarchical data structure of Applicant's invention is enhanced with DATA FLOW relationships that are non-hierarchical, and these non-hierarchical relationships are used to display these non-hierarchical data flow relationships among the machine vision elements that are displayed hierarchically.

The Examiner disagrees with arguments mailed 2/22/05 for three reasons. The first reason is that the Examiner asserts that "Education programs including different application versions such as Paint, Draw, Art Progs, Typing Progs" are similar to "machine vision entities", as claimed by Applicant. Because these

programs are "grouped under the same tree supporting the main software". Using this reasoning, then everything in the world is similar to everything else, as long as they are "grouped under the same tree". This reasoning is clearly erroneous and specious.

By contrast, Applicant asserts that machine vision entities are different from "Education programs", in that Education programs cannot perform any significant machine vision functions. Further, there is no need for "Education programs" to provide or receive "data flow", as is required by Application's invention, as claimed.

The Examiner disagrees that "Habegger is silent on any teaching of Machine Vision", yet this total silence can be readily seen by reference to Figs. 6 and 7A-B. Anyone skilled in the art of machine vision would surely agree, but the Examiner uses his disagreement as the basis for rejection of the claims. Therefore, the Examiner's argument regarding "machine vision entities", or any teaching of "machine vision" in Habegger is deemed to be overcome.

The second reason is that the Examiner disagrees that "Habegger does not teach a plurality of non-hierarchical data flow interrelationships". It's clear to anyone skilled in the art of machine vision, or computer science generally, that there are no "data flow relationships" taught by Habegger. Therefore, it follows logically that there are no "non-hierarchical data flow relationships" taught by Habegger. By contrast, the "non-hierarchical relationships referred to by Habegger in the Summary at col. 1, lines 57-63 are for "cross-linking" nodes in a

first tree to a node in a second tree. This is merely an apparent similarity, since these non-hierarchical relationships are used for an entirely different purpose, and are accordingly not displayed in a user interface, nor do these non-hierarchical relationships have ANYTHING to do with "data flow". Instead, the non-hierarchical relationships in Habegger are used for database searching. However, Applicant does not teach database searching. Database searching has nothing to do with "data flow". In fact, Applicant's invention teaches and claims these major distinctions. Therefore, the Examiner's argument regarding "non-hierarchical data flow relationships" in Habegger is deemed to be overcome.

The third reason is that the Examiner disagrees that "Habegger does not teach display or construction of an enhanced tree-style graphical representation". By reference to Fig. 9, it's clear that Habegger merely teaches how to add a data record to a pre-existing tree, only modifying the contents of the tree, and is totally silent on how to create, construct, or change **the tree structure itself**, as further illustrated by reference to Fig. 10. For example, the first step of Fig. 10 states: "provide a hierarchical data map ...". Habegger is silent on how to create or change a hierarchical data map. Thus, there is NO SUPPORT for the Examiner's assertion that "the map can always change and update to reflect its current condition". In Habegger, the "map" structure does not change. By contrast, Applicant teaches and claims methods that either requires display of the

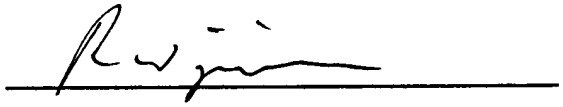
enhanced tree-style graphical representation, or methods that require modifying an enhanced tree-style graphical representation.

Further, the Examiner has not disputed that Habegger does not teach display of the enhanced tree-style graphical representation. Accordingly, the Examiner's argument regarding "Habegger does not teach display or construction of an enhanced tree-style graphical representation" is deemed to be overcome.

Accordingly, Applicants assert that the present application is in condition for allowance, and such action is respectfully requested. The Examiner is invited to phone the undersigned attorney to further the prosecution of the present application.

Respectfully Submitted,

Dated: 9/2/05



Russ Weinzimmer
Registration No. 36,717
Attorney for Applicants

P.O. Box 862
Wilton, NH 03086

Phone: 603-654-5670
Fax: 603-654-3556